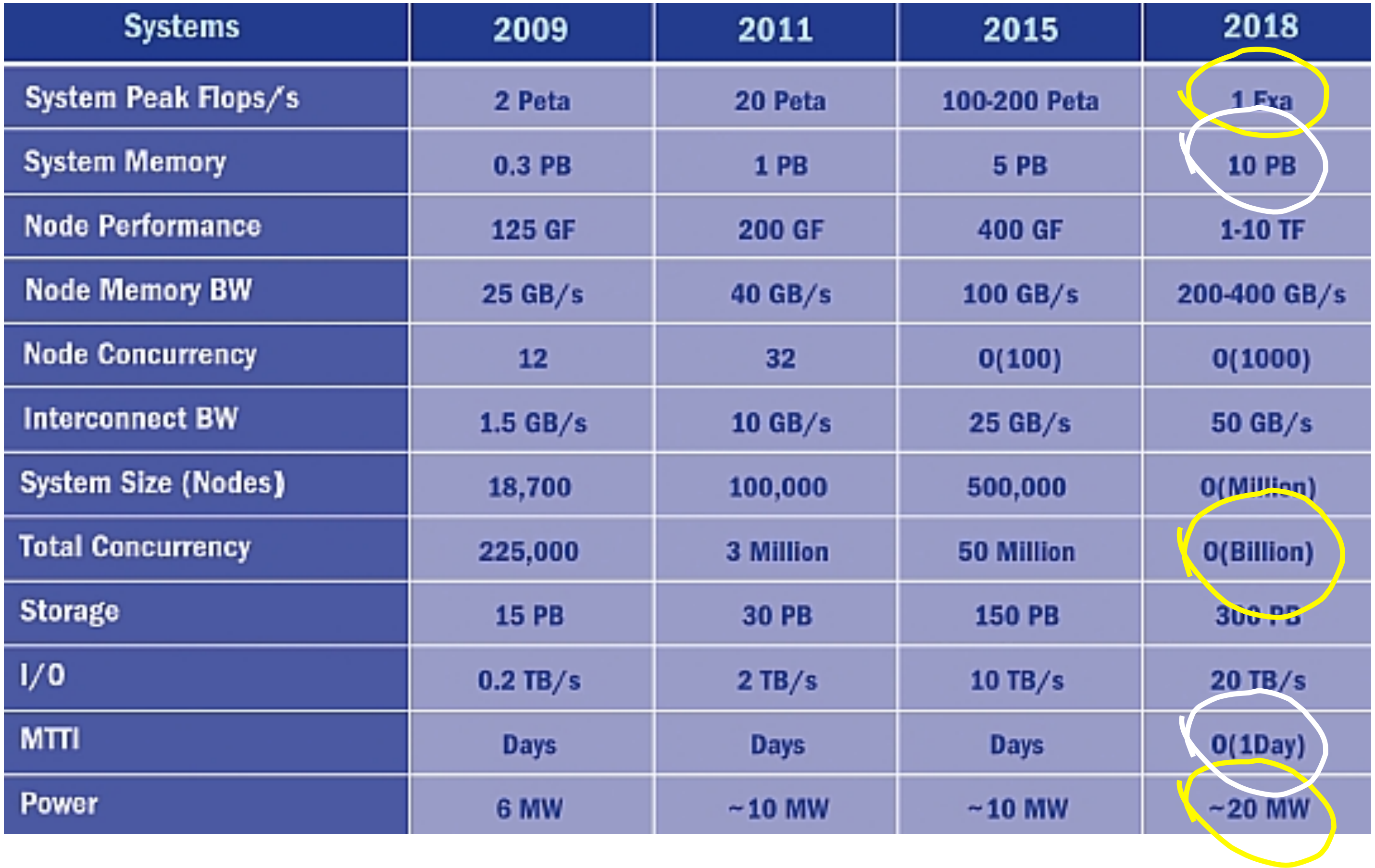
# Modern Computer Architectures …trends

Carlo Cavazzoni – c.cavazzoni@cineca.it
SuperComputing Applications and Innovation Department

# Roadmap to Exascale

(architectural trends)



| Systems | 2009 | 2011 | 2015 | 2018 |
|---|---|---|---|---|
| System Peak Flops/s | 2 Peta | 20 Peta | 100-200 Peta | 1 Exa |
| System Memory | 0.3 PB | 1 PB | 5 PB | 10 PB |
| Node Performance | 125 GF | 200 GF | 400 GF | 1-10 TF |
| Node Memory BW | 25 GB/s | 40 GB/s | 100 GB/s | 200-400 GB/s |
| Node Concurrency | 12 | 32 | O(100) | O(1000) |
| Interconnect BW | 1.5 GB/s | 10 GB/s | 25 GB/s | 50 GB/s |
| System Size (Nodes) | 18,700 | 100,000 | 500,000 | O(Million) |
| Total Concurrency | 225,000 | 3 Million | 50 Million | O(Billion) |
| Storage | 15 PB | 30 PB | 150 PB | 300 PB |
| I/O | 0.2 TB/s | 2 TB/s | 10 TB/s | 20 TB/s |
| MTTI | Days | Days | Days | O(1Day) |
| Power | 6 MW | ~10 MW | ~10 MW | ~20 MW |

# Dennard scaling law (downscaling)

new VLSI gen.

old VLSI gen.

$L' = L / 2$

$V' = V / 2$

$F' = F * 2$

$D' = 1 / L^2 = 4D$

$P' = P$

do not hold anymore!

$L' = L / 2$

$V' = \sim V$

$F' = \sim F * 2$

$D' = 1 / L^2 = 4 * D$

$P' = 4 * P$

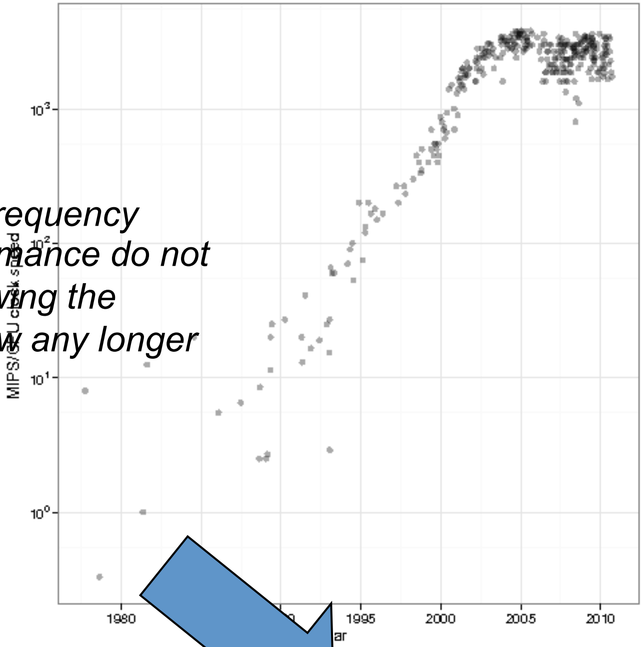
The power crisis!

- Now, power and/or heat generation are the limiting factors of the down-scaling

- Supply voltage reduction is becoming difficult, because Vth cannot be decreased any more, as described later.

- Growth rate in clock frequency and chip area becomes smaller.

*The core frequency and performance do not grow following the Moore's law any longer*



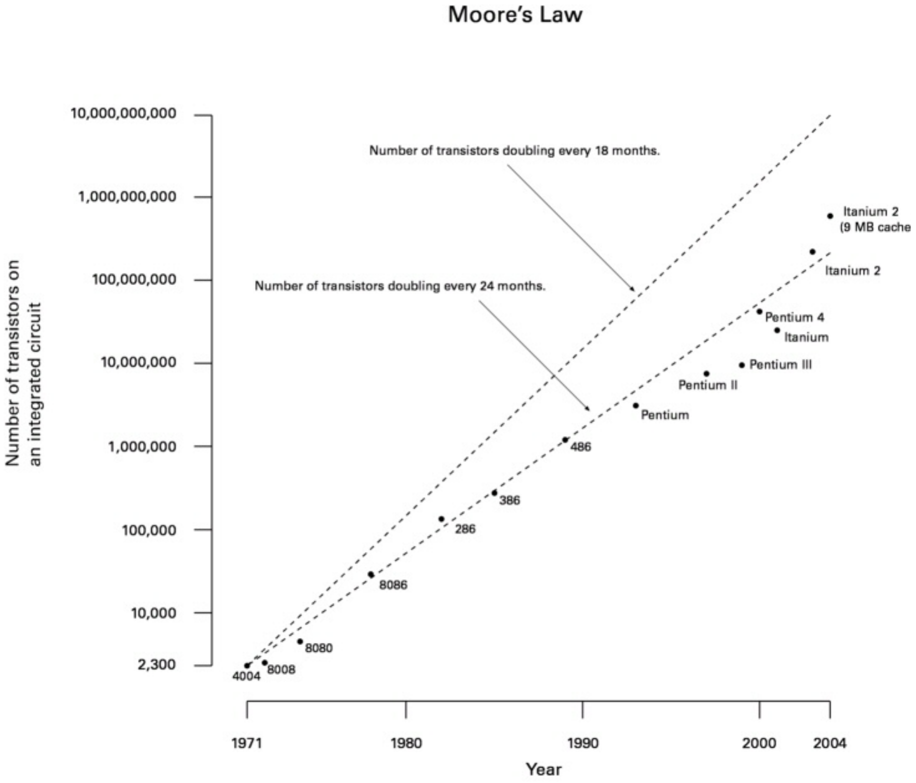
Increase the number of cores to maintain the architectures evolution on the Moore's law

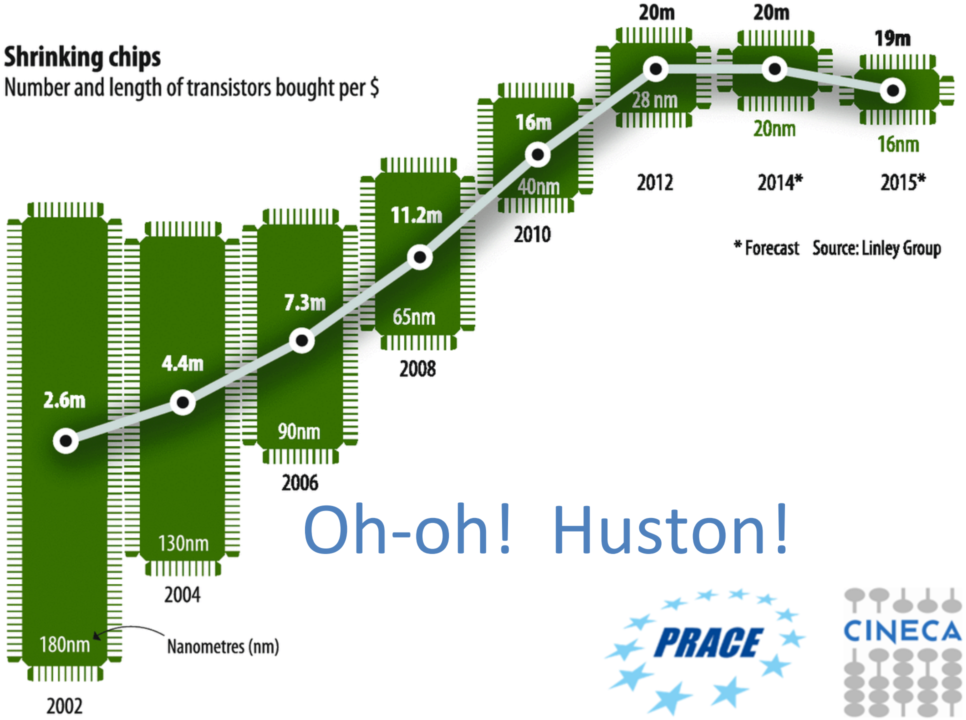
Programming crisis!

# Moore’s Law

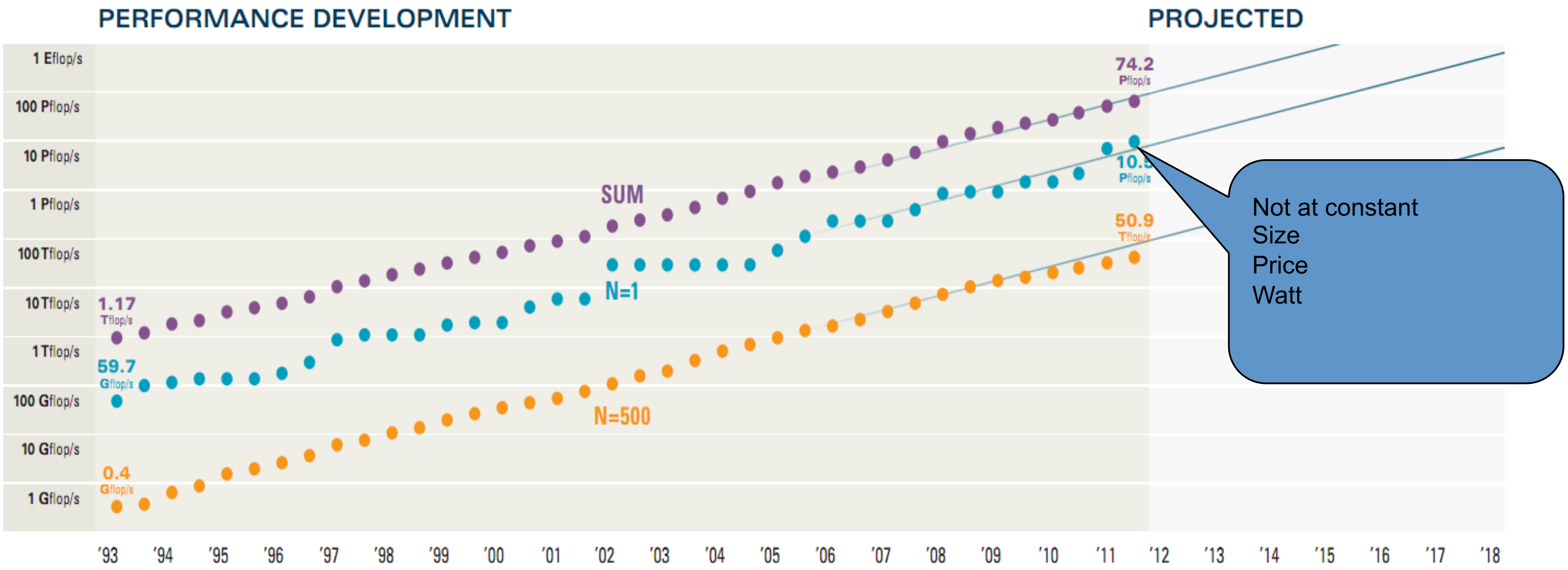
Number of transistors per chip double every 18 month

The true it double every 24 month



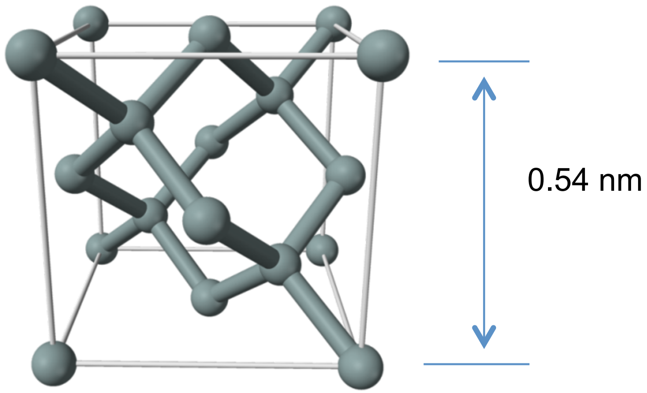


Oh-oh! Huston!

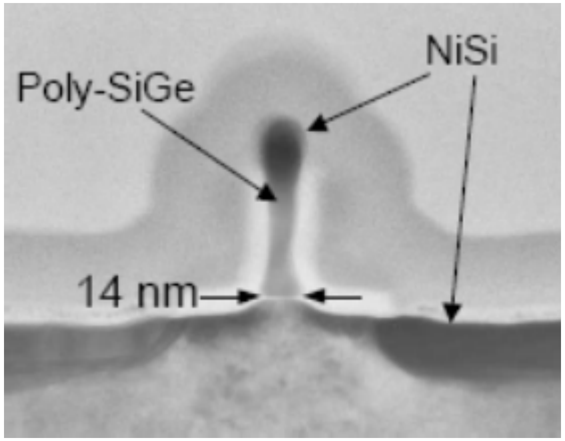


Not at constant
Size
Price
Watt

# The silicon lattice



Si lattice



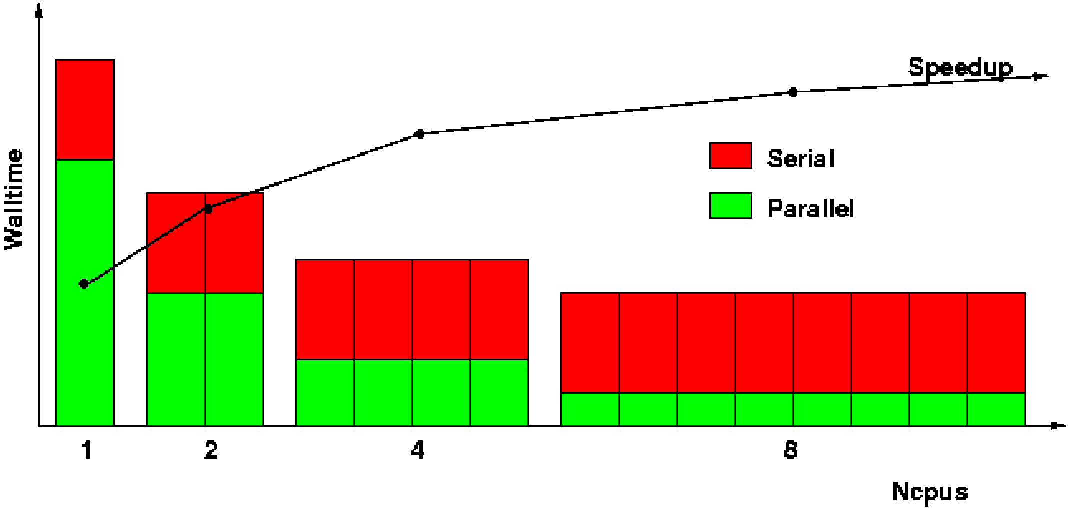
50 atoms!

There will be still 4~6 cycles (or technology generations) left until
we reach 11 ~ 5.5 nm technologies, at which we will reach downscaling limit, in some year between 2020-30 (H. Iwai, IWJT2008).

# Amdahl's law

upper limit for the scalability of parallel applications

determined by the fraction of the overall execution time spent in non-scalable operations (Amdahl's law).



maximum speedup tends to
1 / ( 1 − *P* )
*P*= parallel fraction

1000000 core

*P* = 0.999999

*serial fraction*= 0.000001

# HPC trends

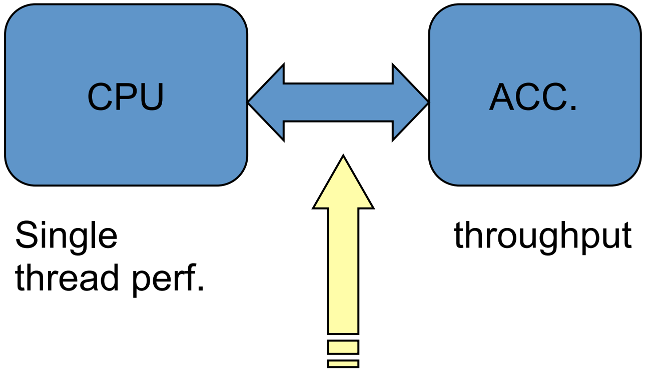
## (constrained by the three law)

| | | | |
|---|---|---|---|
| Peak Performance | ⇧ | exaflops | Moore law |
| FPU Performance | ⇔ | gigaflops | Dennard law |
| Number of FPUs | ⇧ | 10^9 | Moore + Dennard |
| App. Parallelism | ⇧ | Serial fraction<br>1/10^9 | Amdahl's law |

opportunity

challenge

# Architecture toward exascale



CPU

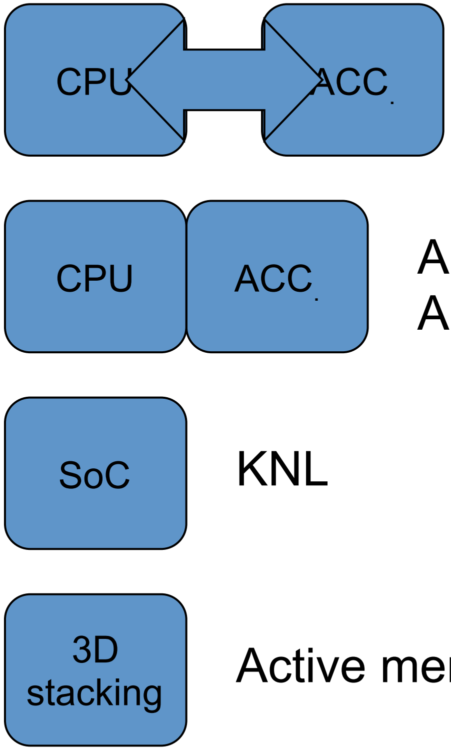
ACC.

GPU/MIC/FPGA

Single
thread perf.

throughput

bottleneck



CPU

ACC.

OpenPower
Nvidia GPU

CPU

ACC.

AMD APU
ARM Big-Little

SoC

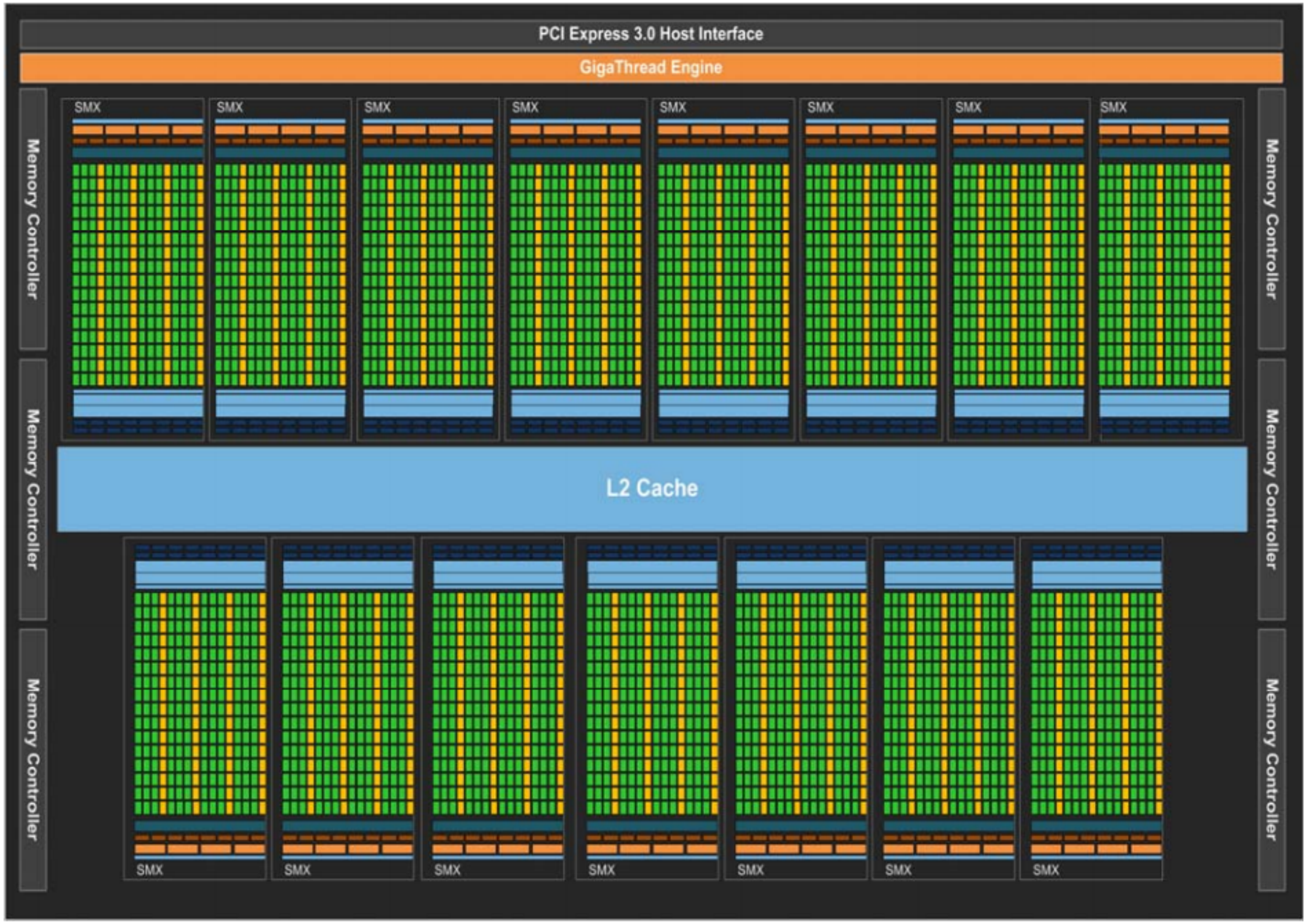
KNL

3D
stacking

Active memory

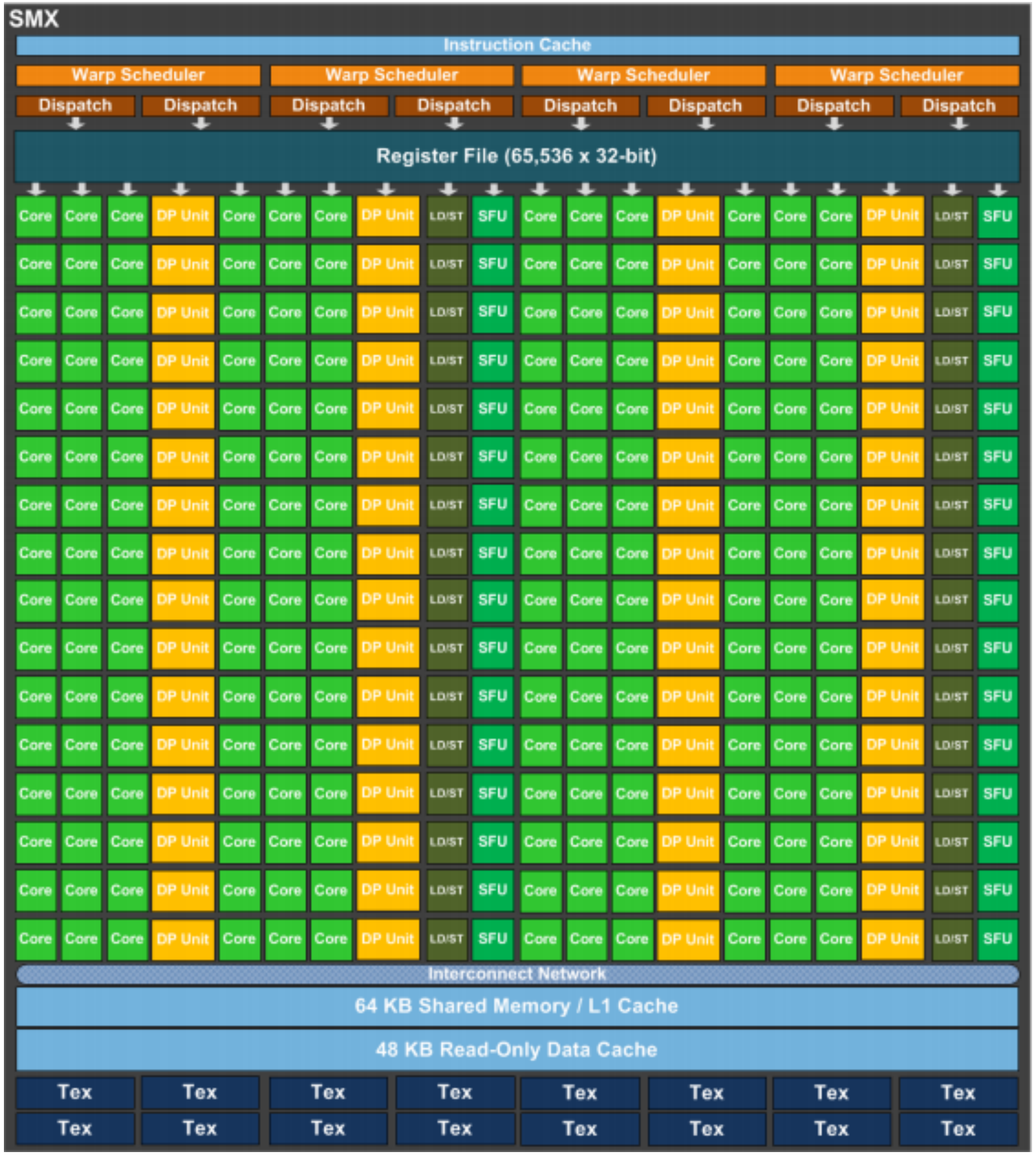
Photonic -> platform flexibility
TSV -> stacking

# K20 nVIDIA GPU



15 SMX Streaming Multiprocessors

# SMX


SMX
Instruction Cache
Warp Scheduler Warp Scheduler Warp Scheduler Warp Scheduler
Dispatch Dispatch Dispatch Dispatch Dispatch Dispatch Dispatch Dispatch
Register File (65,536 x 32-bit)
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU Core Core Core DP Unit Core Core Core DP Unit LD/ST SFU
Interconnect Network
64 KB Shared Memory / L1 Cache
48 KB Read-Only Data Cache
Tex Tex Tex Tex Tex Tex Tex Tex
Tex Tex Tex Tex Tex Tex Tex Tex

192 single precision cuda cores

64 double precision units

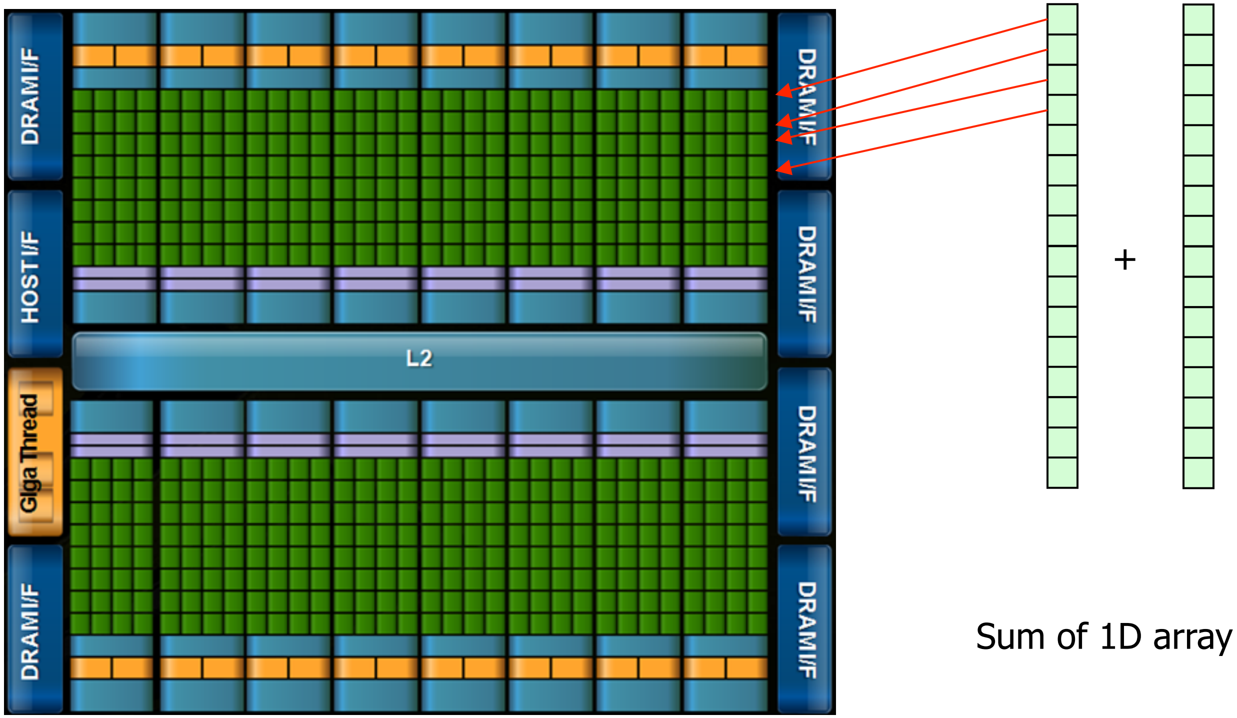
32 special function units

32 load and store units

4 warp scheduler
(each warp contains 32 parallel
Threads)

2 indipendent instruction per warp

# Accelerator/GPGPU



Sum of 1D array

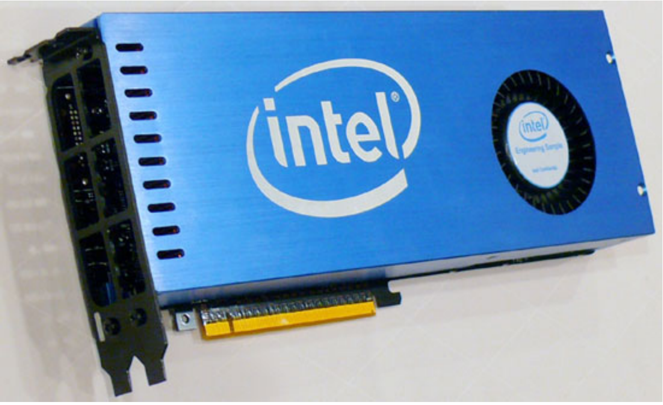
# CUDA sample

```
 void  CPUCode( int* input1, int* input2, int* output, int length) {
                 for ( int  i = 0; i < length; ++i ) {
                       output[ i ] = input1[ i ] + input2[ i ];
                }
 }
```

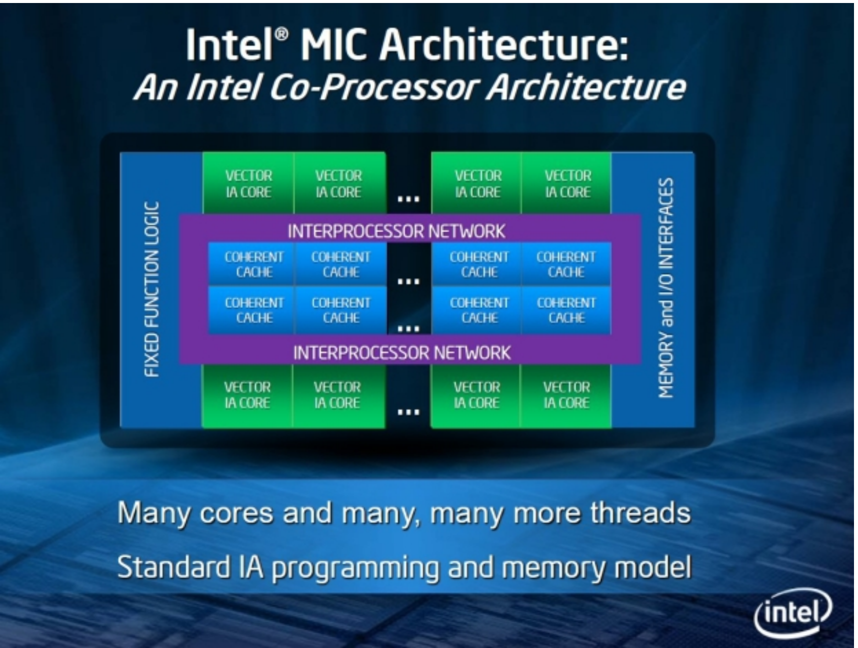
```
__global__void  GPUCode( int* input1, int*input2, int* output, int length) {
               int idx = blockDim.x * blockIdx.x + threadIdx.x;
                if ( idx < length ) {
                      output[ idx ] = input1[ idx ] + input2[ idx ];
               }
}
```

Each thread execute one loop iteration

# Intel MIC



Up to 61 Intel® Architecture cores
1.1 GHz
244  threads
Up to 8 GB memory
up to 352 GB/s bandwidth
512-bit SIMD instructions
Linux* operating system, IP addressable
Standard programming languages and tools
Over 1 TeraFlop/s double precision peak performance



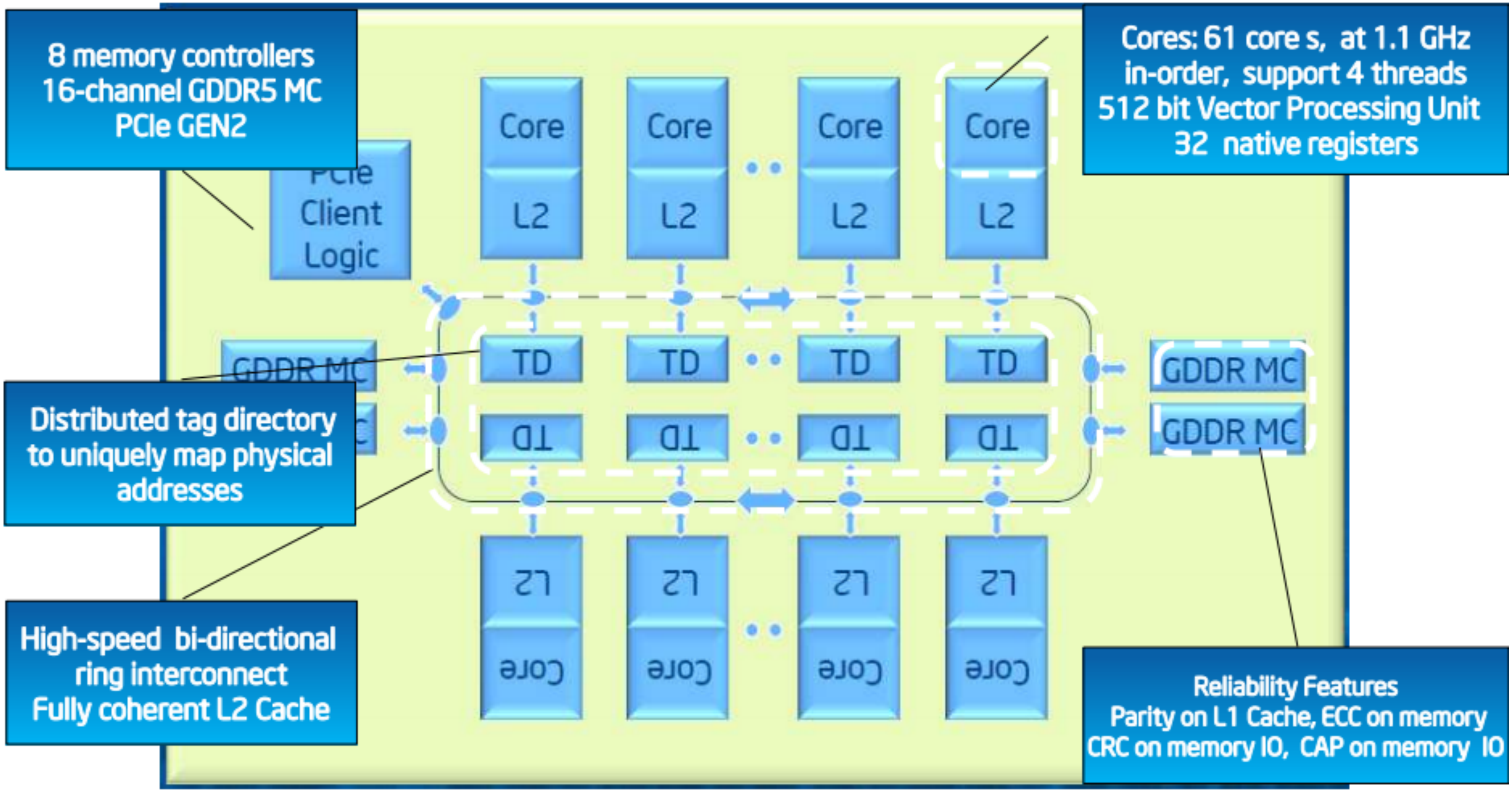
Intel® MIC Architecture:

*An Intel Co-Processor Architecture*

Many cores and many, many more threads

Standard IA programming and memory model

# MIC Architecture



8 memory controllers
16-channel GDDR5 MC
PCIe GEN2

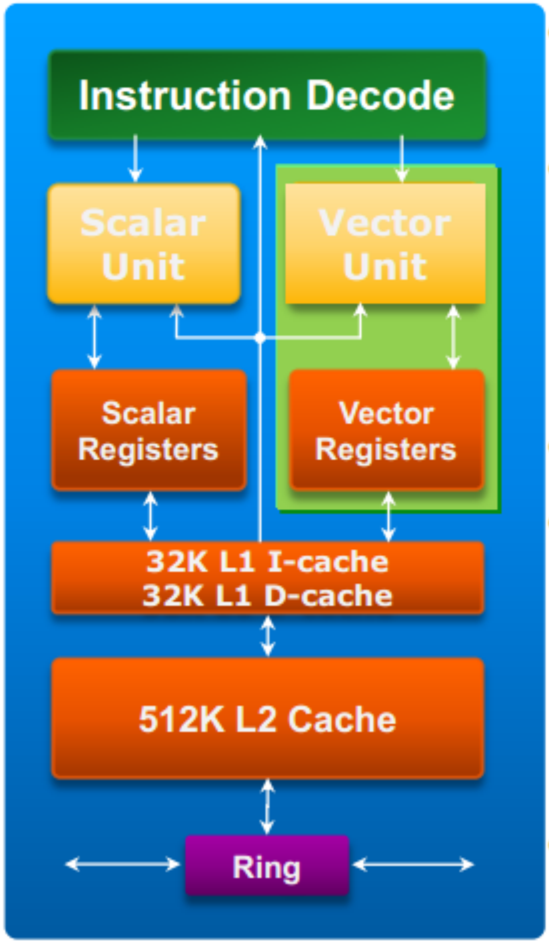
Cores: 61 core s, at 1.1 GHz
in-order, support 4 threads
512 bit Vector Processing Unit
32 native registers

Distributed tag directory to uniquely map physical addresses

High-speed bi-directional ring interconnect
Fully coherent L2 Cache

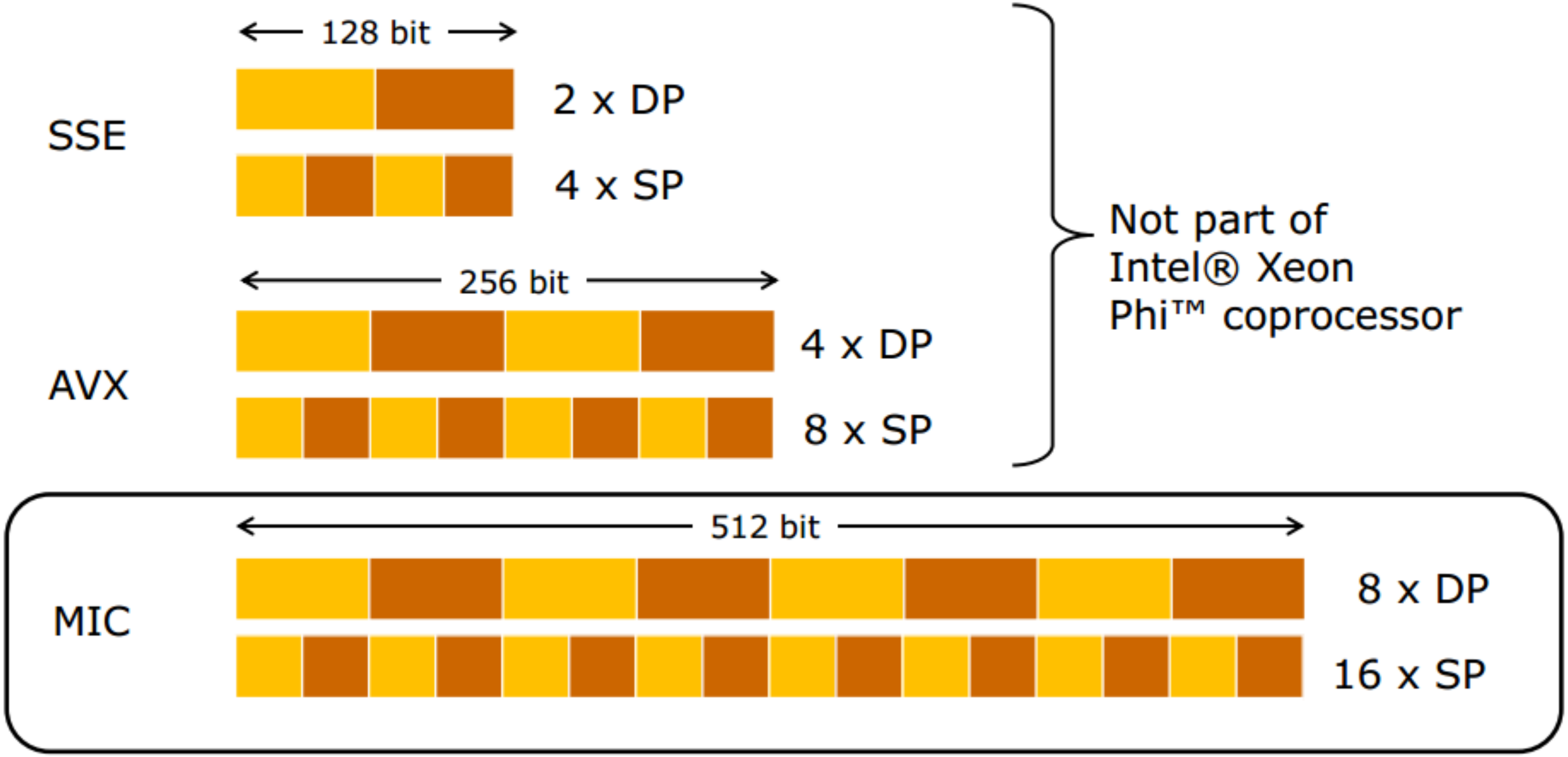
Reliability Features
Parity on L1 Cache, ECC on memory
CRC on memory IO, CAP on memory IO

# Core Architecture



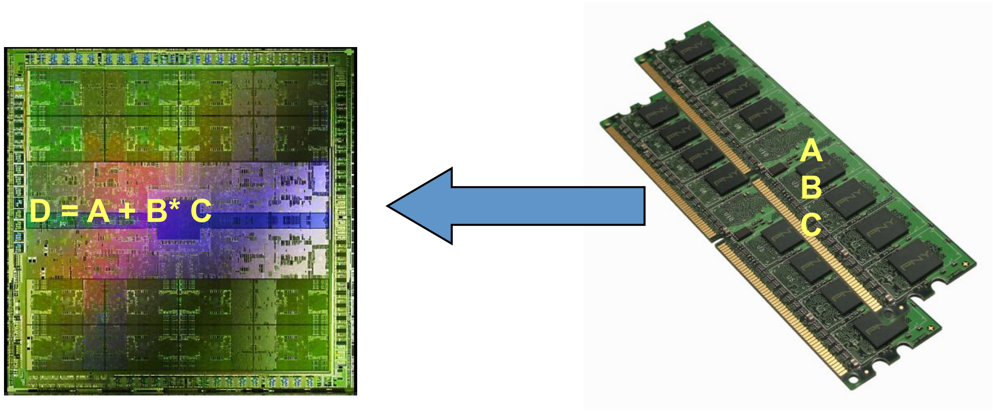
- 60+ in-order, low-power Intel® Architecture cores in a ring interconnect
- Two pipelines
  - Scalar Unit based on Pentium® processors
  - Dual issue with scalar instructions
  - Pipelined one-per-clock scalar throughput
- SIMD Vector Processing Engine
- 4 hardware threads per core
  - 4 clock latency, hidden by round-robin scheduling of threads
  - Cannot issue back-to-back inst in same thread
- Coherent 512 KB L2 Cache per core

# Intel Vector Units



# Memory

Today (at 40nm) moving 3 64bit operands to compute a 64bit floating-point FMA takes 4.7x the energy with respect to the FMA operation itself



Extrapolating down to 10nm integration, the energy required to move date
Becomes 100x !

We need locality! → Fewer memory per core

# Chip Architecture

Strongly market driven → Mobile, Tv set, Screens
Video/Image processing

Intel

ARM

NVIDIA

Power

AMD

→ New arch to compete with ARM
Less Xeon, but PHI

→ Main focus on low power mobile chip
Qualcomm, Texas inst. , Nvidia, ST, ecc
new HPC market, server maket

→ GPU alone will not last long
ARM+GPU, Power+GPU

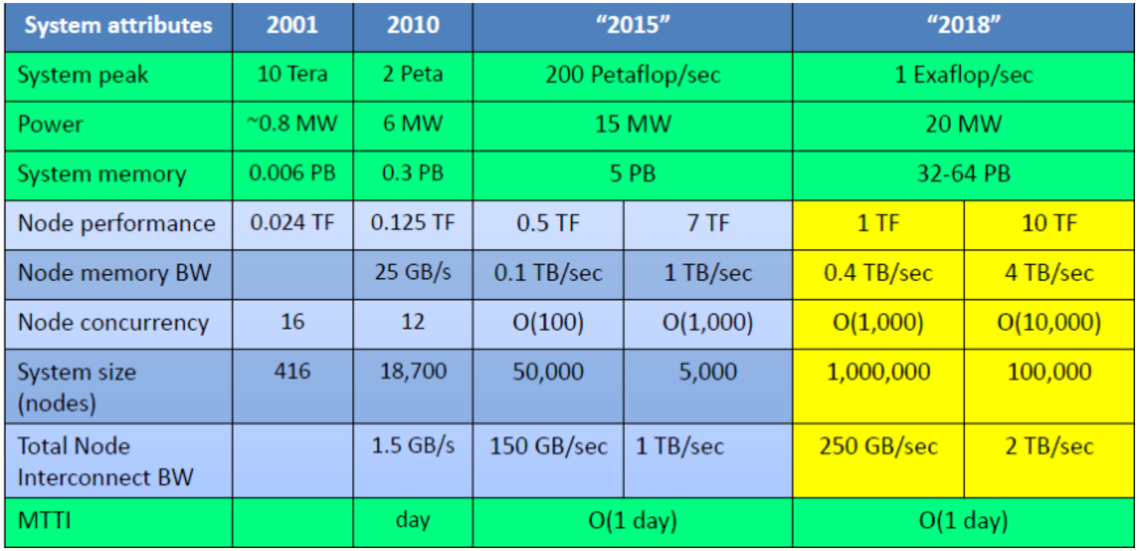
→ Embedded market
Power+GPU, only chance for HPC

→ Console market
Still some chance for HPC

# System architecture

still two models

- Hybrid, but…
  - symmetric
  - asymmetric (a la deep)
- Homogeneus, but…
  - SoC
  - Commodity (a la Mont-Blac)



| System attributes | 2001 | 2010 | "2015" | | "2018" | |
|---|---|---|---|---|---|---|
| System peak | 10 Tera | 2 Peta | 200 Petaflop/sec | | 1 Exaflop/sec | |
| Power | ~0.8 MW | 6 MW | 15 MW | | 20 MW | |
| System memory | 0.006 PB | 0.3 PB | 5 PB | | 32-64 PB | |
| Node performance | 0.024 TF | 0.125 TF | 0.5 TF | 7 TF | 1 TF | 10 TF |
| Node memory BW | | 25 GB/s | 0.1 TB/sec | 1 TB/sec | 0.4 TB/sec | 4 TB/sec |
| Node concurrency | 16 | 12 | O(100) | O(1,000) | O(1,000) | O(10,000) |
| System size (nodes) | 416 | 18,700 | 50,000 | 5,000 | 1,000,000 | 100,000 |
| Total Node Interconnect BW | | 1.5 GB/s | 150 GB/sec | 1 TB/sec | 250 GB/sec | 2 TB/sec |
| MTTI | | day | O(1 day) | | O(1 day) | |

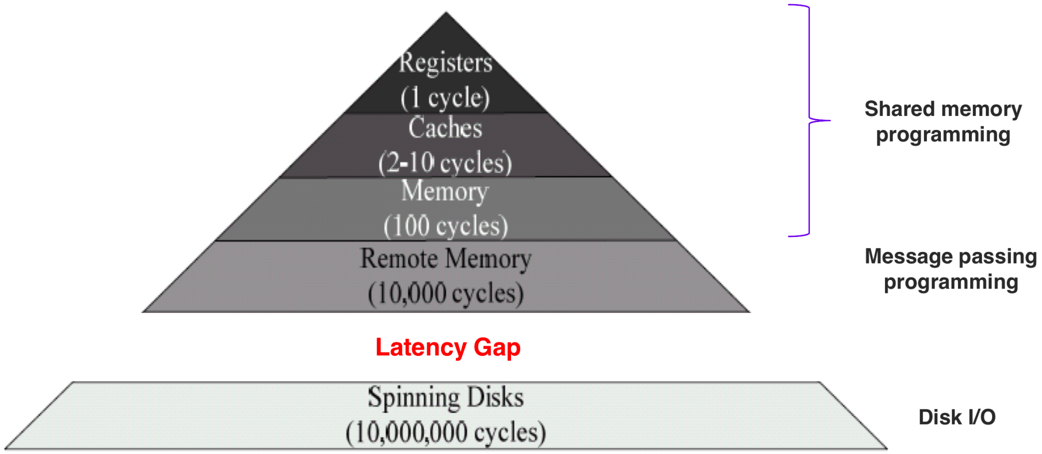
# I/O Challenges

| Today | Tomorrow |
|---|---|
| 100 clients | 10K clients |
| 1000 core per client | 100K core per clients |
| 3PByte | 1Exabyte |
| 3K Disks | 100K Disks |
| 100 Gbyte/sec | 100TByte/sec |
| 8MByte blocks | 1Gbyte blocks |
| Parallel Filesystem | Parallel Filesystem |
| One Tier architecture | Multi Tier architecture |

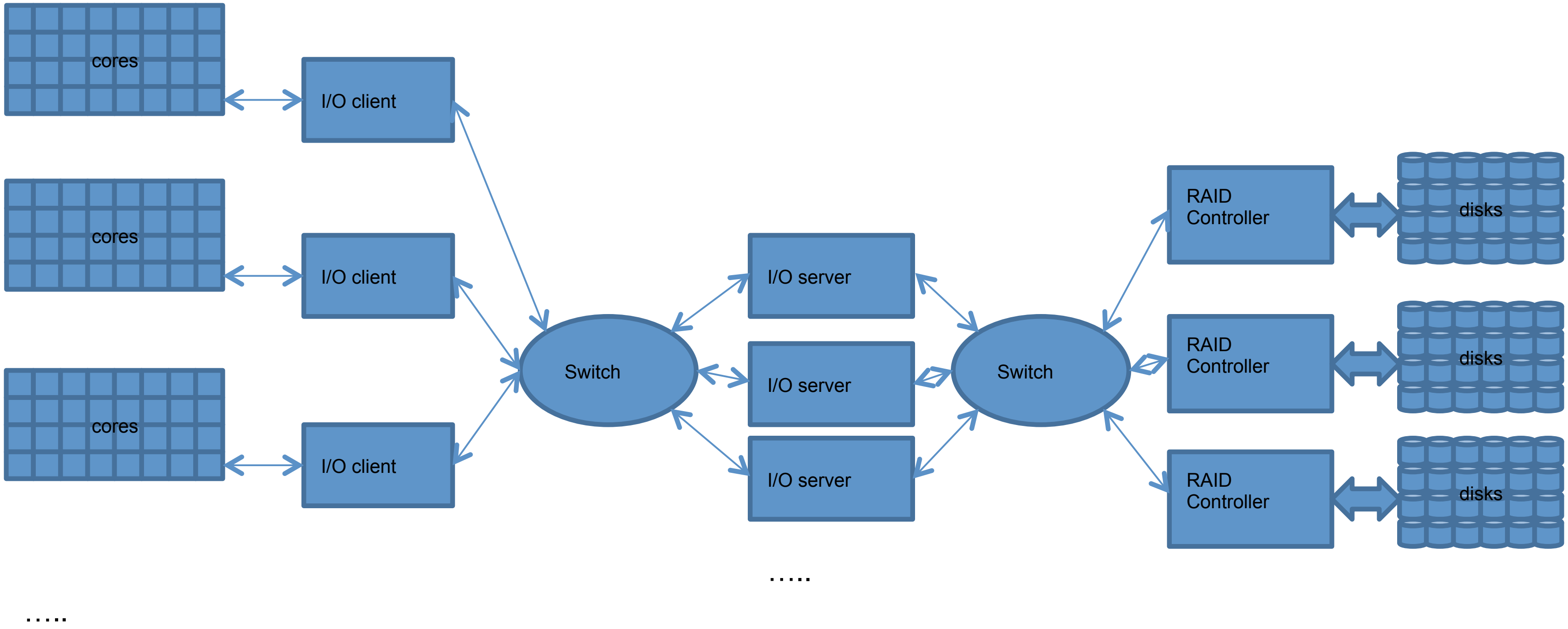
I/O subsystem of high performance computers are still deployed using spinning disks, with their mechanical limitation (spinning speed cannot grow above a certain regime, above which the vibration cannot be controlled), and like for the DRAM they eat energy even if their state is not changed. Solid state technology appear to be a possible alternative, but costs do not allow to implement data storage systems of the same size. Probably some hierarchical solutions can exploit both technology, but this do not solve the problem of having spinning disks spinning for nothing.

# Storage I/O

- The I/O subsystem is not keeping the pace with CPU
- Checkpointing will not be possible
- Reduce I/O
- On the fly analysis and statistics
- Disk only for archiving
- Scratch on non volatile memory ("close to RAM")



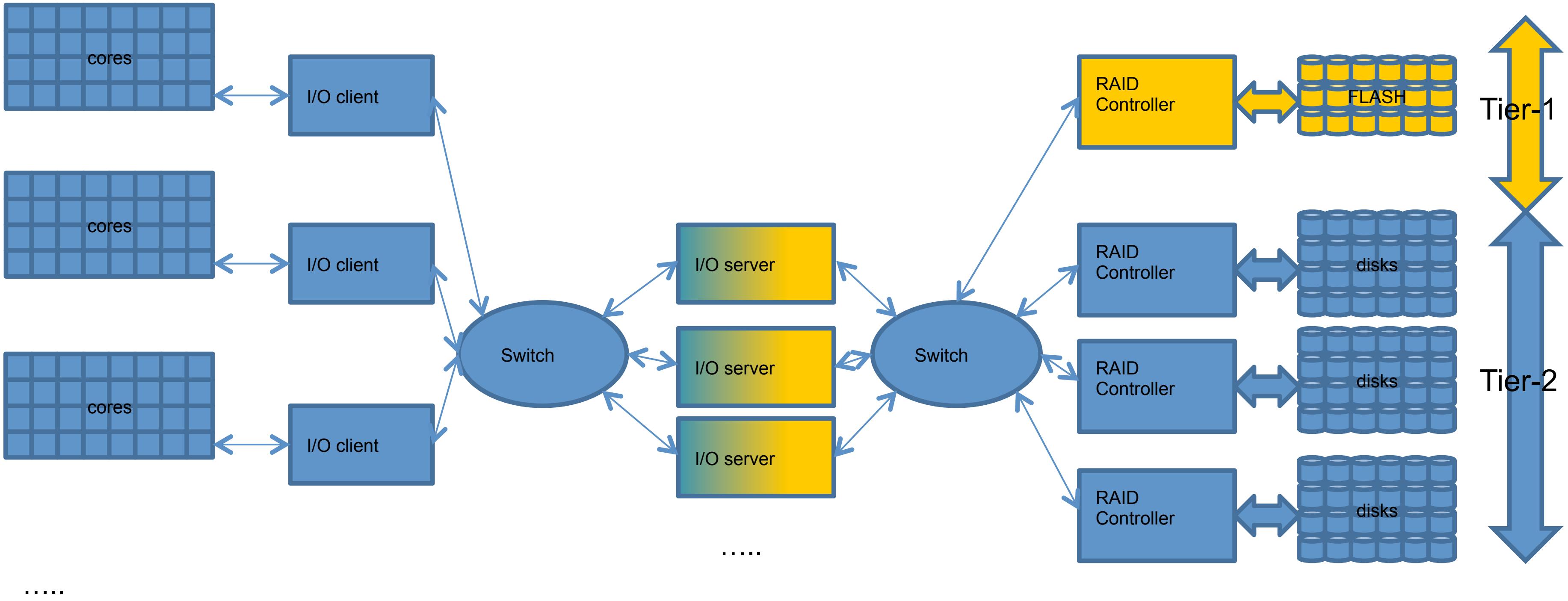
# Today



160K cores, 96 I/O clients, 24 I/O servers, 3 RAID controllers

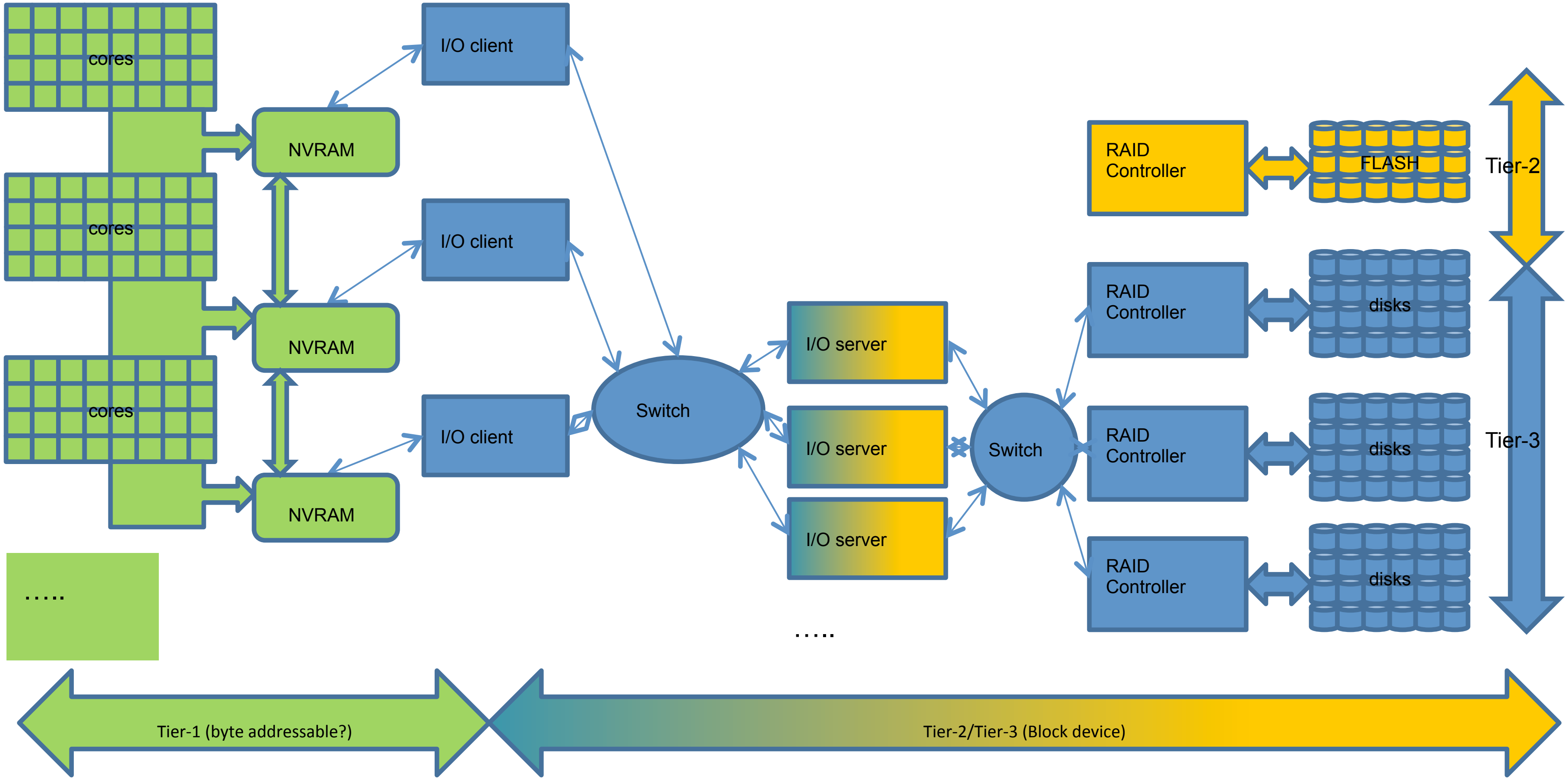
IMPORTANT: I/O subsystem has its own parallelism!

# Today-Tomorrow



1M cores, 1000 I/O clients, 100 I/O servers, 10 RAID FLASH/DISK controllers

# Tomorrow



1G cores, 10K NVRAM nodes, 1000 I/O clients, 100 I/O servers, 10 RAID controllers

# Energy Awareness/Efficiency

# EURORA
# PRACE Prototype experience

**Address Today HPC Constraints:**
- Flops/Watt,
- Flops/m2,
- Flops/Dollar.

**Efficient Cooling Technology:**
- hot water cooling (free cooling);
- measure power efficiency, evaluate (PUE & TCO).

**Improve Application Performances:**
- at the same rate as in the past (~Moore's Law);
- new programming models.

**Evaluate Hybrid (accelerated) Technology:**
- Intel Xeon Phi;
- NVIDIA Kepler.

**Custom Interconnection Technology:**
- 3D Torus network (FPGA);
- evaluation of accelerator-to-accelerator communications.

64 compute cards

128 Xeon SandyBridge (2.1GHz, 95W and 3.1GHz, 150W)

16GByte DDR3 1600MHz per node

160GByte SSD per node

1 FPGA (Altera Stratix V) per node

IB QDR interconnect

3D Torus interconnect

128 Accelerator cards (NVIDA K20 and INTEL PHI)

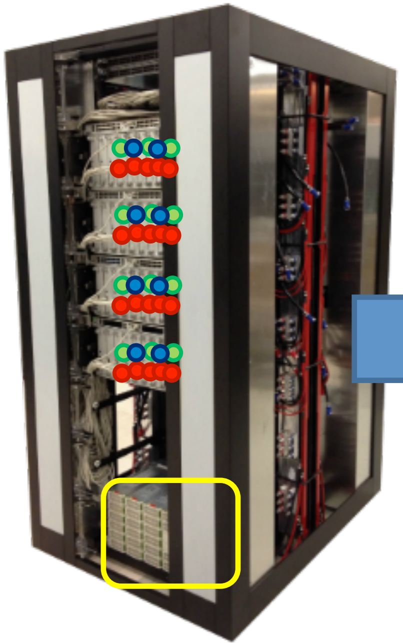
**3,200MOPS/W – 30KW**

**#1 in The Green500 List June 2013**

# Monitoring Infrastructure

Data collection “*front-end*”

- powerDAM (LRZ)
  - Monitoring, Energy accounting
- Matlab
  - Modelling and feature extraction

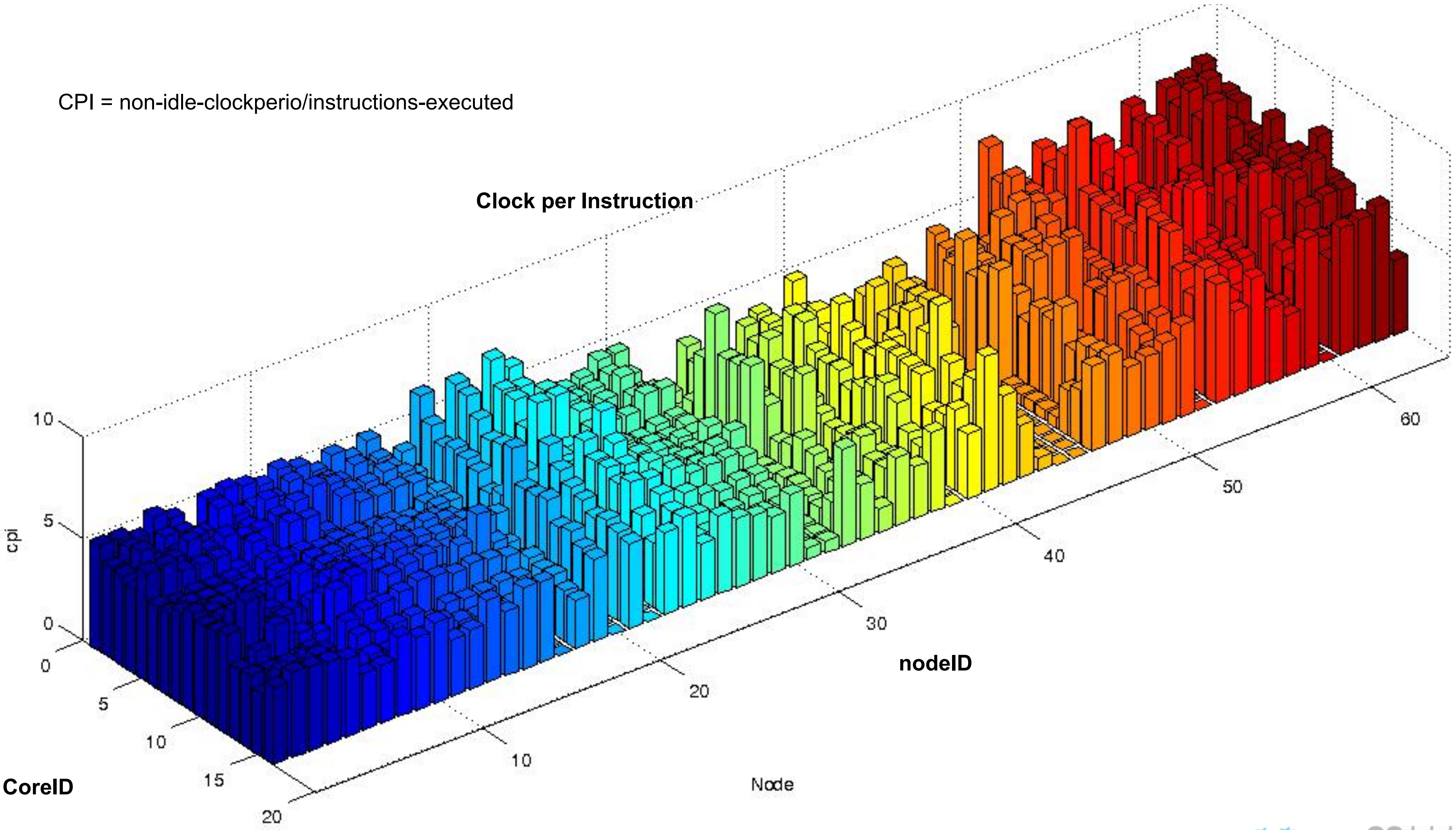


Data collection “*back-end*”

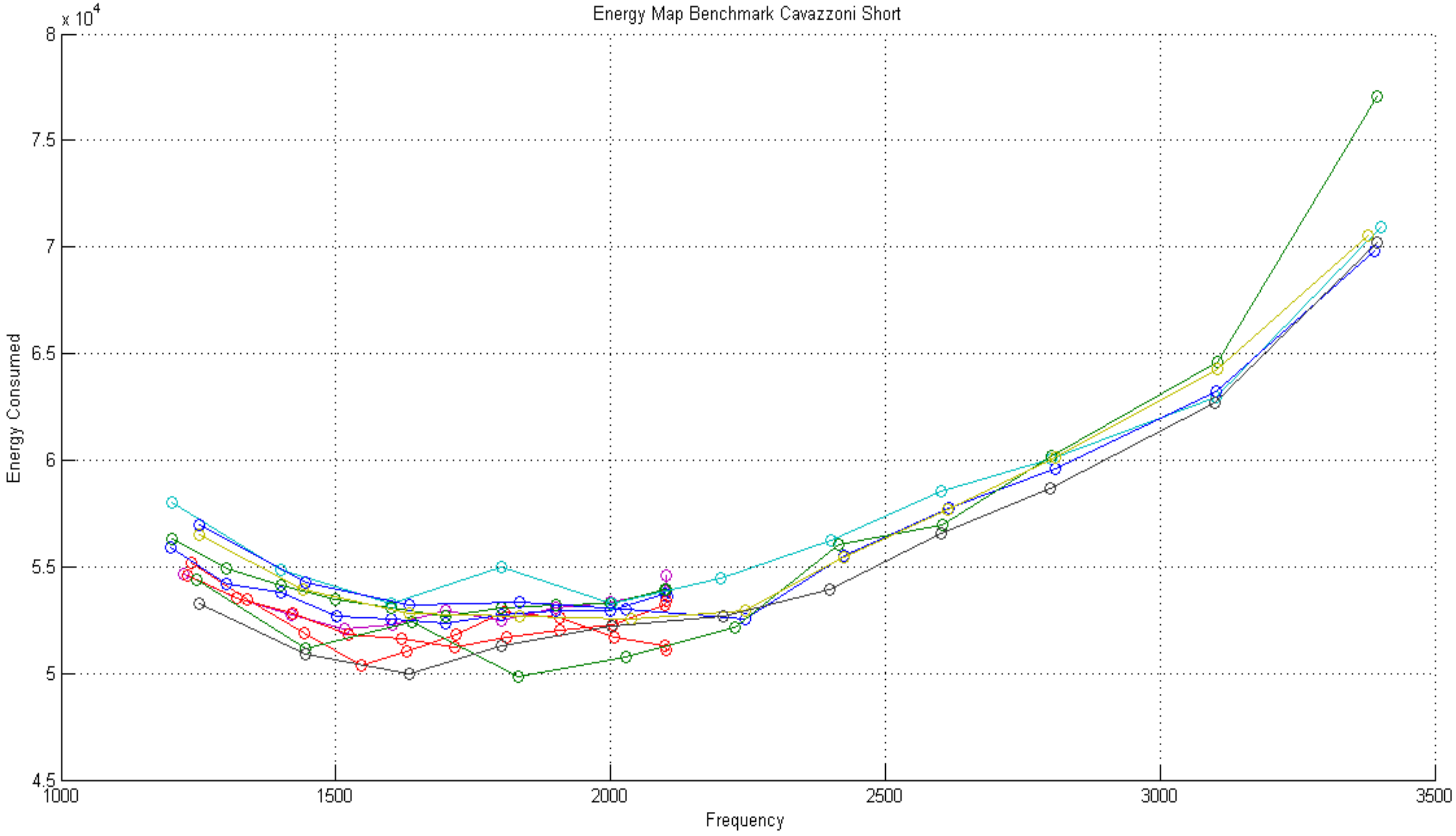
- Node stats (Intel CPUs, Intel MIC, NVidia GPUs)
  - 12-20ms overhead, update every 5s.
- Rack stats (Power Distribution Unit)
- Room stats (Cooling and power supply)
- Job stats (PBS)
- Accounting

# Eurora At Work

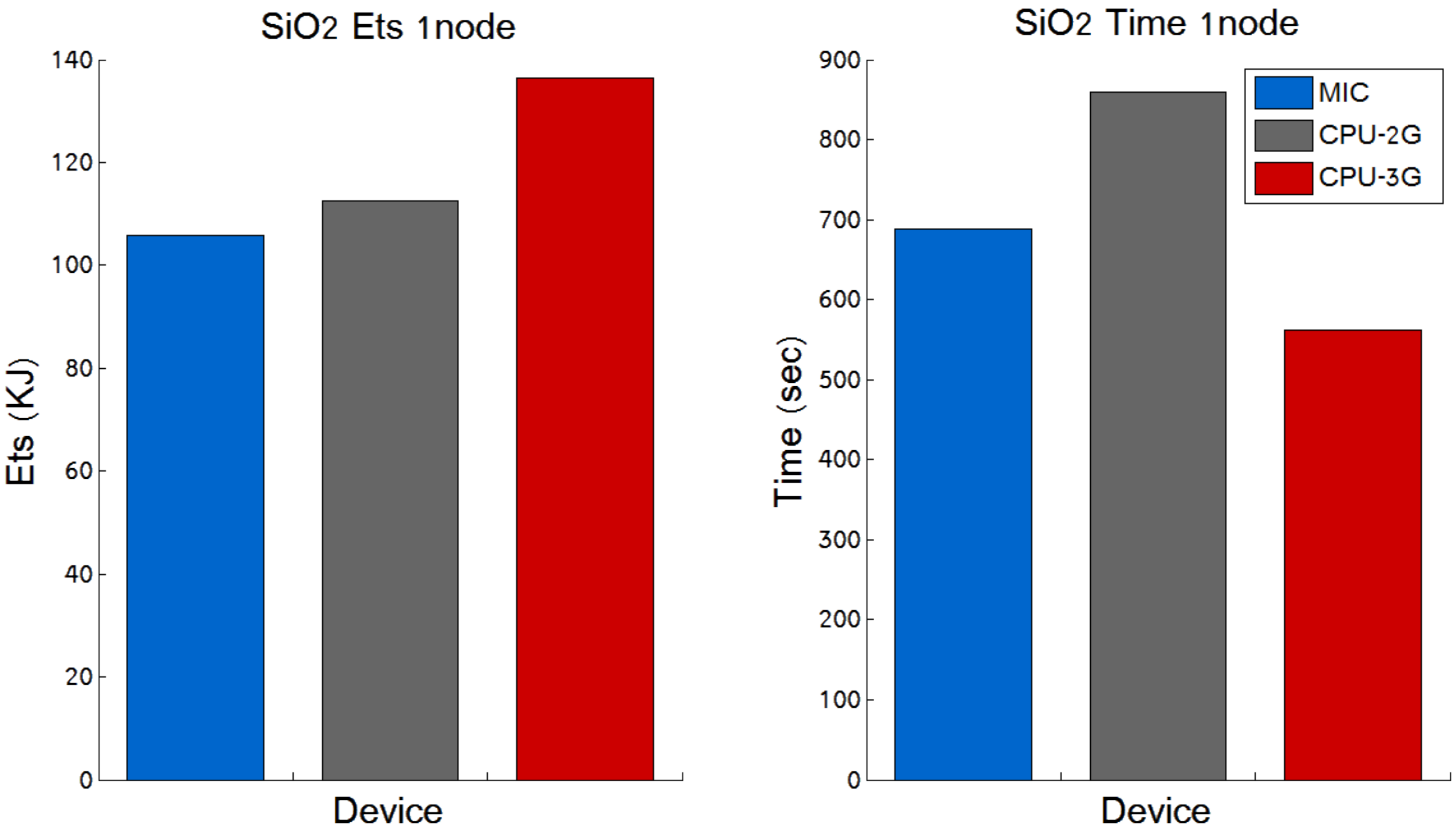
CPI = non-idle-clockperio/instructions-executed



# QE (Al2O3 small benchmark)
# Energy to solution – as a function of the clock

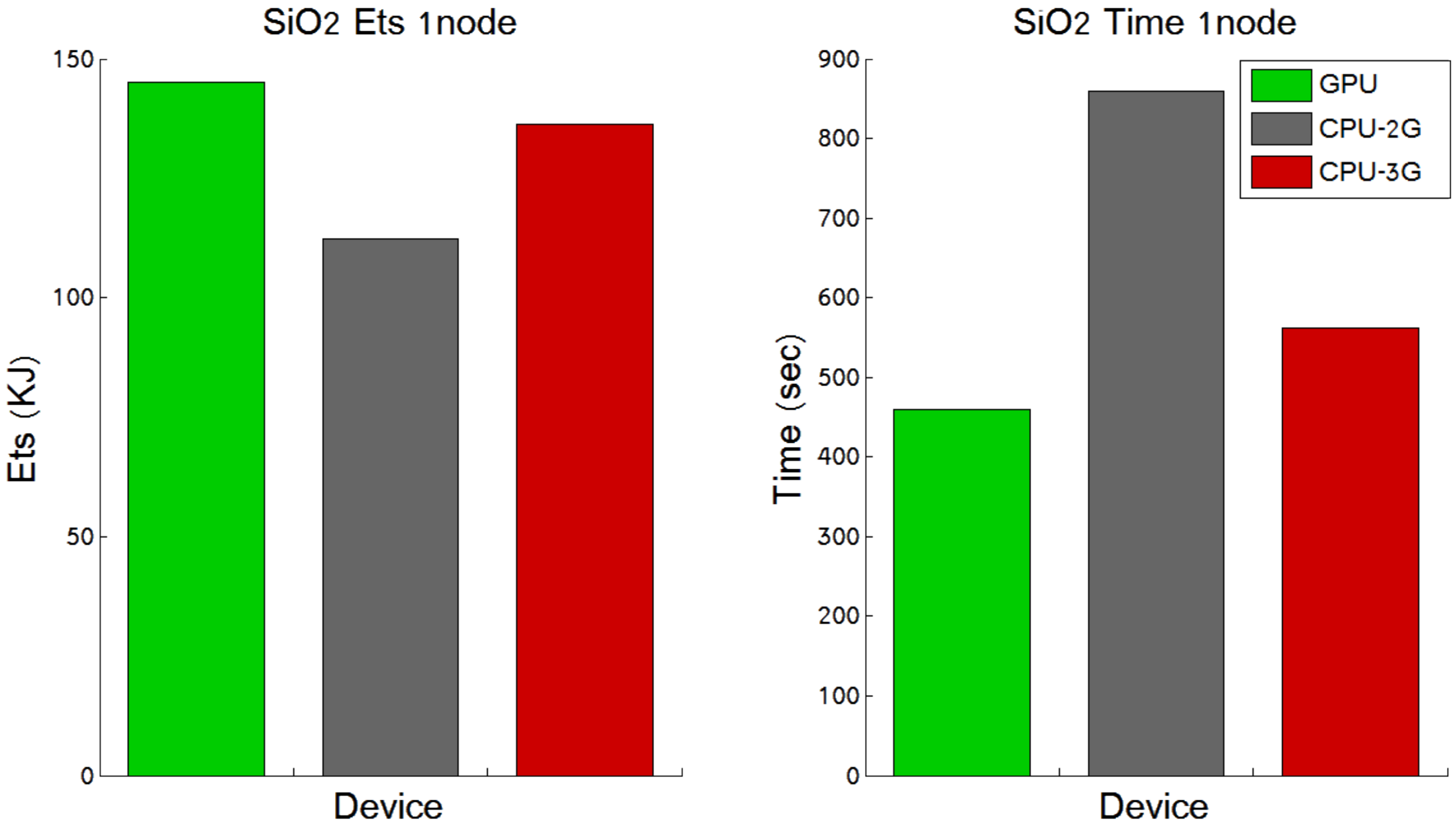


# Quantum ESPRESSO Energy to Solution (PHI)

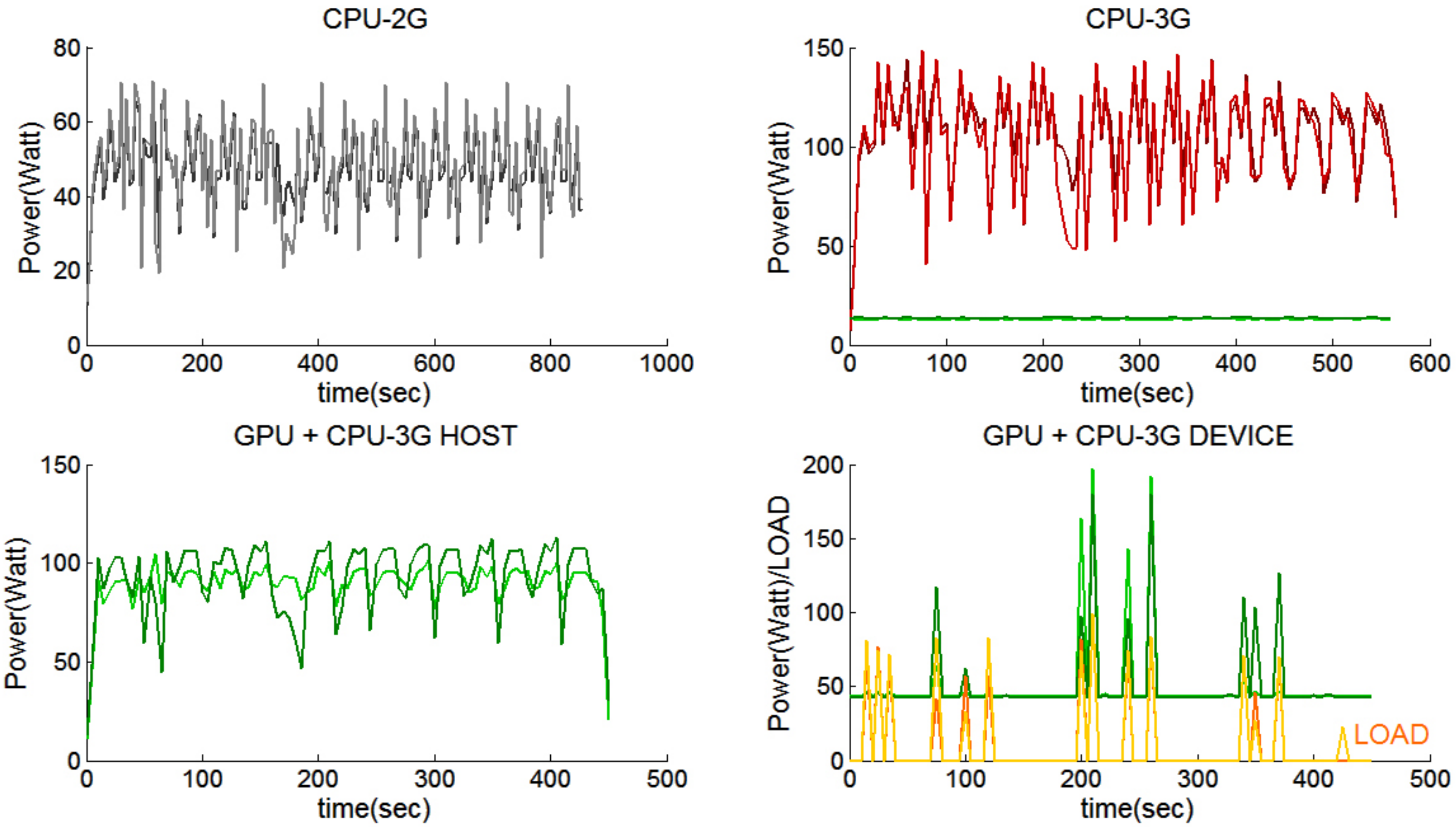


*Time-to-solution (right) and Energy-to-solution (left) compared between Xeon Phi and CPU only versions of QE on a single node.*

# Quantum ESPRESSO Energy to Solution (K20)



*Time-to-solution (right) and Energy-to-solution (left) compared between GPU and CPU only versions of QE on a single node*



CPU-2G

CPU-3G

GPU + CPU-3G HOST

GPU + CPU-3G DEVICE